Pressure Sensor Calibration Method

IPRE100V01 and ICP100V02

About this document

Pressure calibration procedure for the Barometric Pressure Sensor assembled using MEMS die IPRE100V01 and the ASIC die ICP100V02.

Scope and purpose

This document covers the general description of the setup used for pressure calibration, the calibration flow and the registers map.

Intended audience

This document is for Pressure Sensor bare die customers

Table of contents

Table of contents

1 Introduction

This application note is describing the pressure calibration procedure for the Barometric Pressure Sensor assembled using MEMS die IPRE100V01 and the ASIC die ICP100V02. It is supposed the MEMS die and the ASIC die are properly co-packaged to offer good electrical connectivity and mechanical stress decoupling. The pressure calibration method and the setup described in this document can be used as example and adjusted accordingly with the customer needs.

2 Packaging of MEMS and ASIC

For best system performance we recommend to use a shielded package (metal or metal coat with connection to ground). The MEMS die should be decoupled from the mechanical stress caused by the package or PCB as good as possible. A soft glue with a Young's modulus of less than 0.5MPa is recommended. The bond line thickness below the MEMS should be higher than 50um. All materials in the package should have a glass transition temperature that is higher than the operating temperatures, in order to avoid persistent tensions.

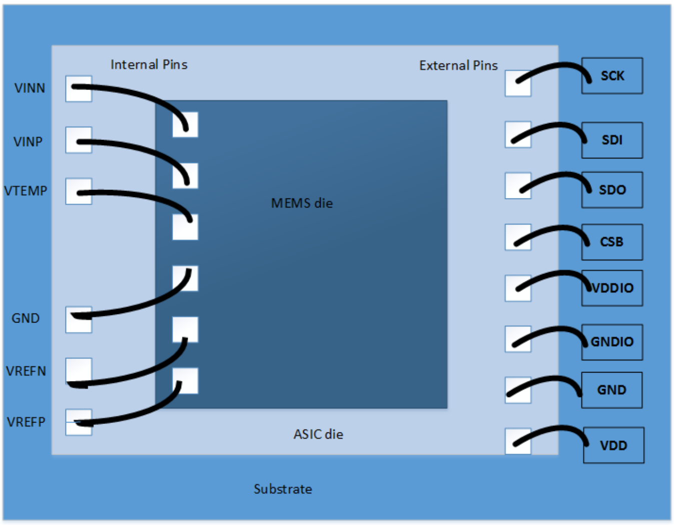


Figure 1 MEMS and ASIC configuration

3 Pressure Measurement and Calibration Setup Diagram

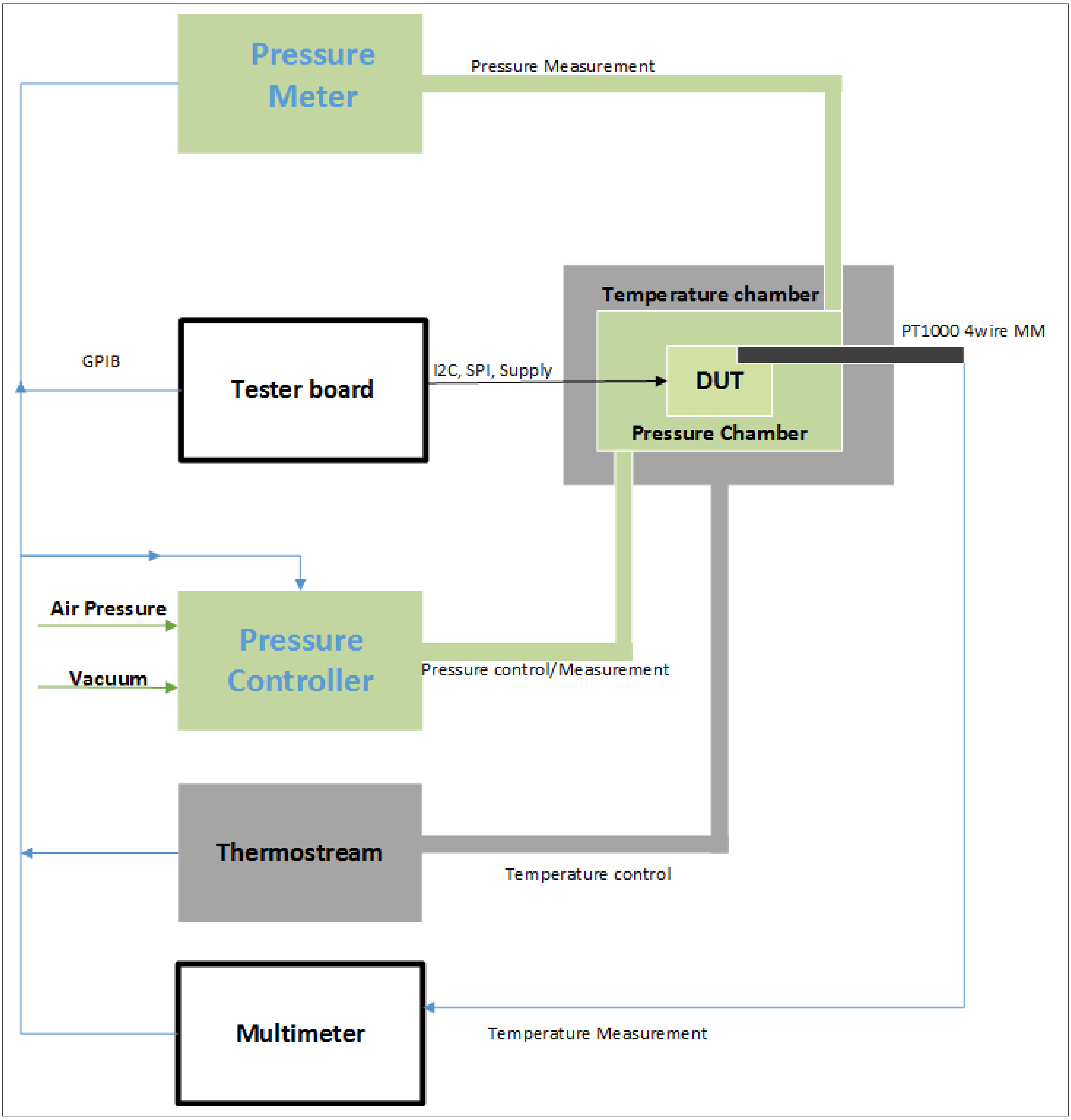


Figure 2 Pressure Measurement and Calibration Setup

4 Pressure Calibration flow

The basic calibration sequence consist of:

- ADC buffer offset trimming which is setting the ADC output in its middle range for the middle pressure operation range (Poffset). The optimal offset value (XXXXX) will be recorded and finally it will be fused;
- ADC buffer gain trimming which is setting the max values of the ADC for minimum value of the pressure operation range. The optimal gain value (YYYY) will be recorded and finally it will be fused.
- Record ADC output value for differed pressures and different temperatures (BE_ptrim) , calculate the coefficients (Poly31)
- The Fuse sequence is fusing the values into OTP's. the "0" value can be overwritten with "0" or "1"

The full sequence and also the detailed description of the Pressure, Temperature and electrical settings are described in the next chapter.

The Voltage values for VDD and VDDIO mentioned in this chapter are given as example and can have any value mentioned as operational voltage in the ICP100V02 datasheet. When fusing, the VDD value must be 4.1V like mentioned in the BE_fuse sub-chapter.

4.1 Pressure Calibration flow diagram

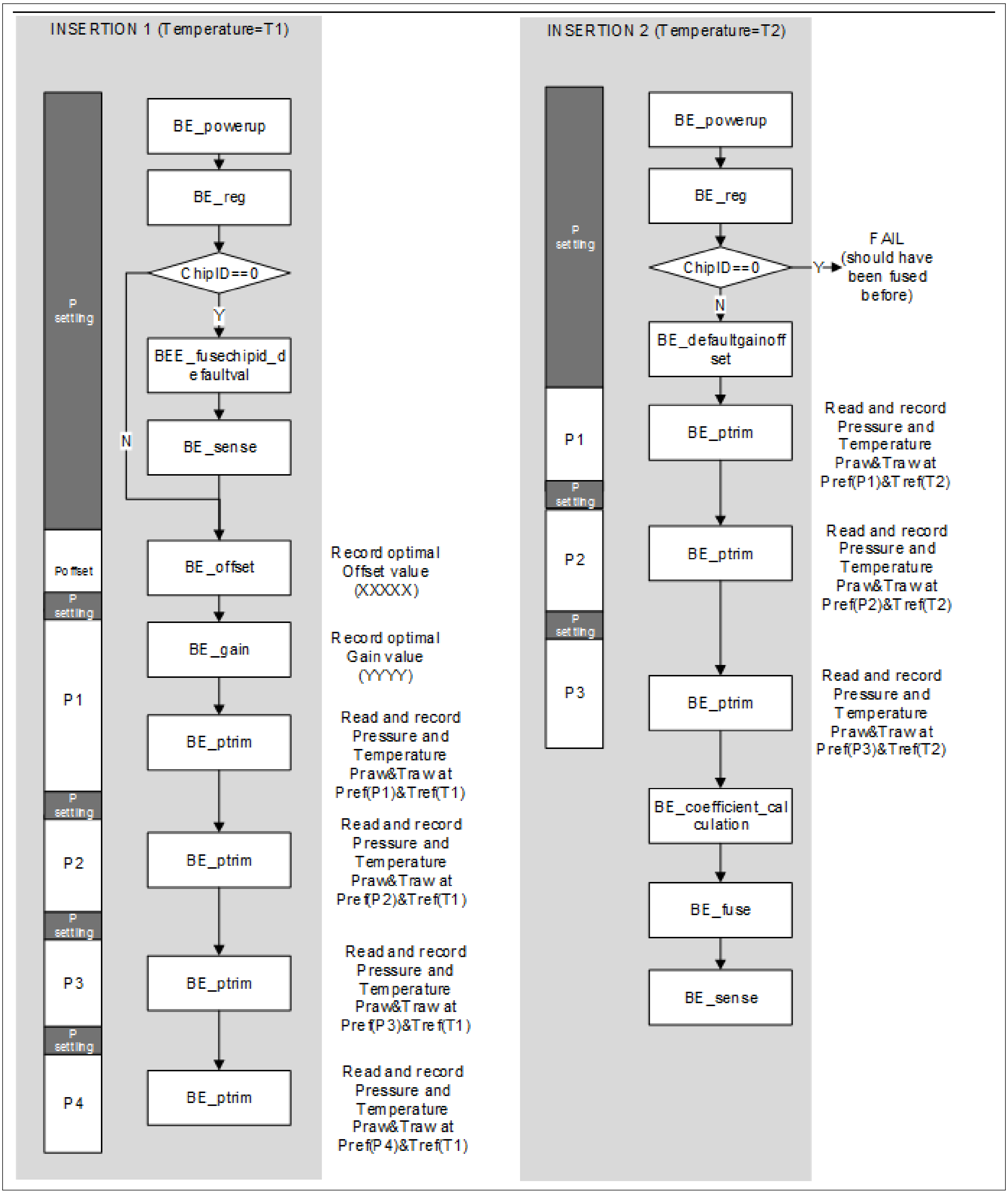


Figure 3 Pressure Calibration flow diagram

4.1.1 Pressure and Temperature settings

Pressure and Temperature settings example

used T and P			
T1=	25	degC	
T2=	60	degC	
P1=	400	hPa	
P2=	850	hPa	
P3=	950	hPa	
P4=	1050	hPa	
Poffset	P2	hPa	for offset
Pgain	P1	hPa	for gain

4.1.2 BE_power_up

This sequence describes the power up of the DUT, Pressure and Temperature settings, when doing Pressure calibration.

P controller: any (the user is free to choose the Pressure value)

T controller: any (the user is free to choose the Temperature value)

I2C frequency [100kHz]

Pattern length: 60.15ms

I2C address : 0x77

Connect at tester: VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

SCL to I2C master

SDA/SDI to I2C master

CSB open

SDO open

Table 1 Power up sequence

Power	**SW extra**	**Chip Control**			
		description	R/W	Address	Data
Power up					
control VDD=1.8V					
control VDDIO=1.8V					
wait>40ms					
	check if correct power up; if "0xC0" cannot be read, repeat for 40ms, if then not read the chip is defect		R	0x08	expect: 0xC0
		read chip ID			
		store Chip ID in database (use with data of etrim and P/T measurements)	R	0x25	Chip ID<30:23>
			R	0x26	Chip ID<22:15>
			R	0x27	Chip ID<14:7>
			R	0x28	Chip ID<6:9>
		Read IDDmin, <1uA expected			

4.1.3 Register access test

This sequence describes the registers involved in Pressure calibration read/write checking via I2C interface.

P controller: any

Temperature: any

I2C frequency: 100.00 [kHz]

Pattern length: 1.1[ms]

I2C address = 0x77

connect at tester: VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

SCL to I2C master

SDA/SDI to I2C master

CSB open

SDO open

Table 2 BE_reg

Power	**SW extra**	**Chip control**			
		description	R/W	Address	Data
control VDD=1.8V					
Control VDD=1.8V					
		check register access			
		write signature 1	W	0x0E	0xA5
		write signature 2	W	0x0F	0x96
		this checks the chip access with I2C	W	0x40 - 0x50	0xAA
			R	0x40 - 0x50	=0xAA?
			W	0x40 - 0x50	0x55
			R	0x40 - 0x50	=0x55?
			W	0x40 - 0x50	0x00
			R	0x40 - 0x50	=0x00?

4.1.4 BE_offset

This sequence describes the ADC buffer offset trimming procedure, Pressure&Temperature settings and R/W commands to be send via I2C interface.

P controller: Poffset

Temperature: T1

I2C frequency: 100.00 [kHz]

Pattern length: 115.54 [ms]

I2C address = 0x77

connect at tester: VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

SCL to I2C master

SDA/SDI to I2C master

CSB open

SDO open

Table 3 BE_offset

Power	**SW extra**	**Chip control**			
		description	R/W	Address	Data
control VDD=1.8V					
control VDDIO=1.8V					
		P resolution config	W	0x06	0x36
		set P result shift for longer integration (average>2048), No FIFO No T shift	W	0x09	0x04
		normal gain for P and high gain for T	W	0x62	0x02
		default AMP gain and UI trim	W	0x64	0b01100UUU
		Start cont P MM	W	0x08	0x05
		default Bridge offset (XXXXX = 00000) or better starting value and binary search in loop A-B	W	0x65	0b000XXXXX
		read P result (clear P-ready bit)	R	0x00-0x02	
	A	wait>105ms			

Table 3 BE_offset (continued)

Power	SW extra	Chip control			
	change bridge offset in loop A-B until minimum P result found	poll if result available	R	0x08	expect: 0xD5 loop until timeout, if no '0xD5' can be read -> defect!
		read P result	R	0x00-0x02	store MSB/LSB/XLSB convert to 2s complement, search minimum
	B	Bridge offset	W	0x65	0b000XXXXX
		restore found Bridge offset	W	0x65	0b000XXXXX
		Stop P MM	W	0x08	0x00
		read P result, clear the P-ready bit …	R	0x00-0x02	
			R	0x08	expect 0xC0

4.1.5 BE_gain

This sequence describes the procedure for ADC buffer amplification trimming including Pressure and Temperature settings.

T/P measurement

P controller: Pgain

Temperature: T1

I2C frequency 100 [kHz]

Pattern length: 115.49 [ms]

I2C address = 0x77

connect at tester: VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

SCL to I2C master

SDA/SDI to I2C master

CSB open

SDO open

Table 4 BE_gain

Power	**SW extra**	**Chip control**			
		description	R/W	Address	Data
control VDD=1.8V					
control VDDIO=1.8V					
		P resolution config	W	0x06	0x36
		set P result shift for longer integration (average>2048), No FIFO No T shift	W	0x09	0x04
		normal gain for P and high gain for T	W	0x62	0x02
		default AMP gain and UI trim (YYYY may start with 0000 or with a better start value and a binary search in loop A-B)	W	0x64	0bYYYY0111

Table 4 BE_gain (continued)

Power	SW extra	Chip control			
		reload found Bridge offset	W	0x65	0b000XXXXX
		Start cont P MM	W	0x08	0x05
		read P result (clear P-ready bit)	R	0x00-0x02	
	A	wait>105ms			
	change bridge gain in loop A-B until desired P result found (~700000)	poll if result available	R	0x08	expect: 0xD5 loop until timeout, if no '0xD5' can be read -> defect!
		read P result	R	0x00 - 0x02	store MSB/LSB/ XLSB convert to 2s complement
	B	vary AMP gain (YYYY), keep other bits constant	W	0x64	0bYYYY0111
		reload found AMP gain	W	0x64	0bYYYY0111
		Stop P MM	W	0x08	0x00
		read P result, clear the P-ready bit …	W	0x00 - 0x02	
			R	0x08	expect 0xC0

4.1.6 BE_defaultgainoffset

This sequence is describing the settings for ADC buffer gain and offset found in the previous insertion (see "Insertion 1" from the Pressure calibration flow diagram).

P controler: any

Temperature: T2

I2C frequency 100.00 [kHz]

Pattern length 0.06 [ms]

I2C address = 0x77

connect at tester: VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

CSB open

SDO open

Table 5 BE_defaultgainoffset

Power	**SW extra**	**Chip control**			
		description	R/W	Address	Data
control VDD=1.8V					
control VDDIO=1.8V		default values			
		reload bridge offset	W	0x65	0b000XXXXX
		reload found AMP gain	W	0x64	0bYYYY0111

4.1.7 BE_ptrim

This sequence is describing the Pressure trim procedure using four Pressure test points and two temperatures.

P controller P1/P2/P3/P4

Temperature T1/T2

I2C frequency 100.00 8kHz]

Pattern length: 2101.20 [ms]

I2C address = 0x77

connect at tester: VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

SCL to I2C master

SDA/SDI to I2C master

CSB open

SDO open

Table 6 BE_ptrim

Power	SW extra	Chip control			
		description	R/W	Address	Data
control Vdd=1.8V VDDIO=1.8V					
		write signature 1	W	0x0E	0xA5
		write signature 2	W	0x0F	0x96
		P resolution config	W	0x06	0x36
		set P result shift for longer integration (average>2048), no FIFO No T shift	W	0x09	0x04
		T resolution config	W	0x07	0xB0
		normal gain for P and high gain for T	W	0x62	0x02
		Start P/T MM	W	0x08	0x07
wait >200ms					

Table 6 BE_ptrim (continued)

Power	SW extra	Chip control			
		read P and T (continuous access)	R	0x00 - 0x05	data: 24 bit 2's complement P data (from address 0x00-0x02) and 24 bit 2's complement T data (from address 0x03-0x06)
	READ IDDmax, <400uA expected				
		Stop P/T MM	W	0x08	0x00

4.1.8 BE_fusechip_defaultval

This sequence is describing the chip ID fusing procedure.

I2C freq 100.00 [kHz]

Pattern length: 115.18 [ms]

I2C address = 0x77

connect at tester: VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

SCL to I2C master

SDA/SDI to I2C master

CSB open

SDO open

Write the fields according below table (write also the '0'), use "Address when fusing"

The chip ID is fused in front-end and all allocated bits must be written 0 to preserve it

Table 7 EFUSE Map

Description	Address when fusing	Address accesible in user space	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	0x40	0x10	0							
	0x41	0x11	0							
	0x42	0x12	0							
	0x43	0x13	0							
	0x44	0x14	0							
	0x45	0x15	0							
	0x46	0x16	0							
	0x47	0x17	0							
	0x48	0x18	0							
	0x49	0x19	0							
	0x4A	0x1A	0							
	0x4B	0x1B	0							
	0x4C	0x1C	0							
	0x4D	0x1D	0							
	0x4E	0x1E	0							
	0x4F	0x1F	0							
	0x50	0x20	0							
	0x51	0x21	0							

Table 7 EFUSE Map (continued)

Description	Address when fusing	Address accesible in user space	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	0x52	0x22	0							
	0x53	0x23	0							
	0x54	0x24	0	0	chip ID<37:32>					
	0x55	0x25	chip ID <31:24>							
	0x56	0x26	chip ID <23:16>							
	0x57	0x27	chip ID <15:8>							
	0x58	0x28	0	chip ID <6:0>						
	0x60	0x30	0	1011011						
	0x61	0x31	0	0	0	10110				
	0x62	0x32	0	0	0	0	0	0	1	0
	0x63	0x33	0	1	1	0	0	1	1	0
	0x64	0x34	0	0	0	0	0	1	1	1
	0x65	0x35	0	0	0	0	0	0	0	0
	0x66	0x36	0	0	0	0	0	0	0	0
	0x67	0x37	0	0	0	0	0	0	0	0
	0x68	0x38	0	0	0	0	0	1	1	1
	0x69	0x39	0	0	0	0	0	0	0	0
	0x6F	0x3F	0	0	0	0	0	0	0	0

The chip ID coding is:

Lot ID week (6 bit): ID(37:32)

Lot ID running number (10 bit): ID(31:22)

Wafer number (5 bit): ID(21:17)

Chip X coordinate (8 bit): ID(16:9)

Chip Y coordinate (8 bit): ID(8 & 6:0)

4.1.9 BE_fuse

This sequence describes the fusing procedure for calibration coefficients

I2C frequency 100.00 [kHz]

Pattern length: 115.66 [ms]

I2C address = 0x77

connect at tester VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

SCL to I2C master

SDA/SDI to I2C master

CSB open

SDO open

Write the fields according below table (write also the '0'), use "Address when fusing"

Table 8 Blow efuse coefficients values

Power	**SW extra**	**Chip control**			
		description	R/W	Address	Data
control VDDIO=1.8V					
		write signature 1	W	0x0E	0xA5
		write signature 2	W	0x0F	0x96
		read the 8bit in register 0x31 and store to intermediate variable 'foundOSC'	R	0x31	0x??
		read the 8bit in register 0x33 and store to intermediate variable 'foundLDO'	R	0x33	0x??
		Write found trimming HW/SW values to Efuse shadow	W	0x40 to 0x6F	previous found trimming info write also 0 for not to be fused cells in specific address
control VDD=4.1V	wait for VDD settling 20ms				
		select internal 2kHz clock	W	0x70	0x20

Table 8 Blow efuse coefficients values (continued)

Power	SW extra	Chip control			
		blow command	W	0x70	0x22
	The next two write operations to 0x61 and 0x63 should be performed at 3.2ms after the blow command. If this timing is not possible on the test system, it must be assured that the write command does not happen after 3.2ms, so a wait time shorter than 3.2ms is in this case also OK				
		write 'foundOSC' to register 0x61	W	0x61	'foundOSC'
		write 'foundLDO' to register 0x63	W	0x63	'foundLDO'
	wait until reg 0x70==0x20	poll if blow is ready - expect X==0	R	0x70	0x2X
control VDD=1.8V	wait for VDD settling 20ms				
		normal clocking again	W	0x70	0x00

Table 9 EFUSE Map

Description	Address when fusing	Address accesible in user space	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
	0x40	0x10	C0 <11:4>							
	0x41	0x11	C0<3:0>				C1<11:8>			
	0x42	0x12	C1 <7:0>							
	0x43	0x13	C00 <19:12>							
	0x44	0x14	C00 <11:4>							

4 Pressure Calibration flow

Table 9 EFUSE Map (continued)

<table>
<tr><th>Description</th><th>Address when fusing</th><th>Address accesible in user space</th><th>bit7</th><th>bit6</th><th>bit5</th><th>bit4</th><th>bit3</th><th>bit2</th><th>bit1</th><th>bit0</th></tr>
<tr><td></td><td>0x45</td><td>0x15</td><td colspan="4">C00<3:0></td><td colspan="4">C10<19:16></td></tr>
<tr><td></td><td>0x46</td><td>0x16</td><td colspan="8">C10 <15:8></td></tr>
<tr><td></td><td>0x47</td><td>0x17</td><td colspan="8">C10 <7:0></td></tr>
<tr><td></td><td>0x48</td><td>0x18</td><td colspan="8">C01 <15:8></td></tr>
<tr><td></td><td>0x49</td><td>0x19</td><td colspan="8">C01 <7:0></td></tr>
<tr><td></td><td>0x4A</td><td>0x1A</td><td colspan="8">C11 <15:8></td></tr>
<tr><td></td><td>0x4B</td><td>0x1B</td><td colspan="8">C11 <7:0></td></tr>
<tr><td></td><td>0x4C</td><td>0x1C</td><td colspan="8">C20 <15:8></td></tr>
<tr><td></td><td>0x4D</td><td>0x1D</td><td colspan="8">C20 <7:0></td></tr>
<tr><td></td><td>0x4E</td><td>0x1E</td><td colspan="8">C21 <15:0></td></tr>
<tr><td></td><td>0x4F</td><td>0x1F</td><td colspan="8">C21 <7:0></td></tr>
<tr><td></td><td>0x50</td><td>0x20</td><td colspan="8">C30 <15:8></td></tr>
<tr><td></td><td>0x51</td><td>0x21</td><td colspan="8">C30 <7:0></td></tr>
<tr><td></td><td>0x52</td><td>0x22</td><td colspan="8">0</td></tr>
<tr><td></td><td>0x53</td><td>0x23</td><td colspan="8">0</td></tr>
<tr><td></td><td>0x54</td><td>0x24</td><td colspan="8">0</td></tr>
<tr><td></td><td>0x55</td><td>0x25</td><td colspan="8">0</td></tr>
<tr><td></td><td>0x56</td><td>0x26</td><td colspan="8">0</td></tr>
<tr><td></td><td>0x57</td><td>0x27</td><td colspan="8">0</td></tr>
<tr><td></td><td>0x58</td><td>0x28</td><td>1</td><td colspan="7">0</td></tr>
<tr><td></td><td>0x60</td><td>0x30</td><td>0</td><td colspan="7">0</td></tr>
<tr><td></td><td>0x61</td><td>0x31</td><td>0</td><td>0</td><td>0</td><td colspan="5">0</td></tr>
<tr><td></td><td>0x62</td><td>0x32</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td></td><td>0x63</td><td>0x33</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td></td><td>0x64</td><td>0x34</td><td>Y</td><td>Y</td><td>Y</td><td>Y</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td></td><td>0x65</td><td>0x35</td><td>0</td><td>0</td><td>0</td><td>X</td><td>X</td><td>X</td><td>X</td><td>X</td></tr>
<tr><td></td><td>0x66</td><td>0x36</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td></td><td>0x67</td><td>0x37</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td></td><td>0x68</td><td>0x38</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td></td><td>0x69</td><td>0x39</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td></td><td>0x6F</td><td>0x3F</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>

4.1.10 BE_sense

This sequence is describing the fuse values checking procedure.

Pcontroller: any

Temperature: any

I2C frequency: 100:00 [kHz]

Pattern length: 1.52 [ms]

I2C address = 0x77

connect at tester: VDD to chip supply

VDDIO to chip supply

GND to ground

GND to ground

SCL to I2C master

SDA/SDI to I2C master

CSB open

SDO open

Table 10 BE_sense

Power	SW extra	Chip control			
		description	R/W	Address	Data
control VDD=1.8V control VDDIO=1.8V					
		write signature 1	W	0x0E	0xA5
		write signature 2	W	0x0F	0x96
		reset latch where efuse is sensed to	W	0x0C	0x00
			W	0x0C	0x03
		sense command	W	0x70	0x01

Table 10 BE_sense (continued)

Power	SW extra	Chip control			
		read the sensed values	R	0x10 - 0x3F	compare data to expected (== previously fused) values eg.: reg 0x10 == reg 0x40 reg 0x11 == reg 0x41 …. reg 0x3F == reg 0x6F
		reset latch where efuse is sensed to	W	0x0C	0x00
		reset latch where efuse is sensed to	W	0x0C	0x03
		sense comman d with test margin high	W	0x70	0x05
		read the sensed values	R	0x10-0x3F	compare data to expected (== previously fused) values eg.: reg 0x10 == reg 0x40 reg 0x11 == reg 0x41 …. reg 0x3F == reg 0x6F

4.1.11 BE_coefficients_calculator

4 Pressure Calibration flow

Calculation of Calibration Coefficient

Following the Ptrim procedure described in the previous sections, the user should end up to a number of matrices namely x, y, xt, yt of Pressure and Temperature raw data, as well as Reference Pressure and Temperature data. These matrices will be used for fitting the experimental data to a user defined calibration formula. An example-formula (poly31) is shown below. The fitting process in case of poly31 calibration formula involves 4 pressure and 2 temperature points.

Poly31

$$Pcal = c00 + Praw_{sc} * \left(c10 + Praw_{sc} * (c20 + Praw_{sc} * c30)\right) + Traw_{sc} * c01 +$$

$$+ Traw_{sc} * Praw_{sc} * (c11 + Praw_{sc} * c21);$$

with $Praw_{sc} = \frac{Praw}{kP}$, $Traw_{sc} = \frac{Traw}{kT}$ the normalized raw data and kP and kT the corresponding normalization parameters given in the table below:

Oversampling Rate	Scale Factor (kP or kT)
1 (single)	524288
2 times (Low Power)	1572864
4 times	3670016
8 times	7864320
16 times (Standard)	253952
32 times	516096
64 times (High Precision)	1040384
128 times	2088960

The matrices in such case have the following form:

$$x = \begin{bmatrix} P1(T1)/kP & T1(P1)/kT \\ P2(T1)/kP & T1(P2)/kT \\ P3(T1)/kP & T1(P3)/kT \\ P4(T1)/kP & T1(P4)/kT \\ P1(T2)/kP & T2(P1)/kT \\ P2(T2)/kP & T2(P2)/kT \\ P3(T2)/kP & T2(P3)/kT \end{bmatrix}, y = \begin{bmatrix} Pref1(Tref1) \\ Pref2(Tref1) \\ Pref3(Tref1) \\ Pref4(Tref1) \\ Pref1(Tref2) \\ Pref2(Tref2) \\ Pref3(Tref2) \end{bmatrix}$$

where

$P_i(T_j)$ corresponds to the pressure raw data measured at pressure Prefi and temperature Trefj ,

$T_j(P_i)$ corresponds to the Temperature raw data measured at Pressure Pi and temperature Tj,

and i = 1 to 4 and j = 1 to 2

4 Pressure Calibration flow

The user should keep as a remark that in general the size of the matrices might vary depending on the number of pressure and temperature test points used for the calibration process.

Finally, for the calculation of the cij coefficients, the test data have to be fitted to the formula. To obtain such coefficient estimates, the user can use the linear least squares method since the fitting polynomial is linear. In case of more customized calibration functions the user can instead use nonlinear least squares, linear or piecewise interpolation etc. depending on the complexity of the function. For simplicity we provide below an example of fitting code in Matlab where by default linear least squares method is used:

```
% Fitting

[fit_inv_m,gof_m] = fit(x,y,'poly31')

c00 =fit_inv_m.p00;
c10 =fit_inv_m.p10;
c20 =fit_inv_m.p20;
c30 =fit_inv_m.p30;
c01 =fit_inv_m.p01;
c11 =fit_inv_m.p11;
c21 =fit_inv_m.p21;
```

The "round" function rounds the coefficients to the nearest integer neighbor before these are being stored to the corresponding ASIC registers.

```
% Quantisation

c00q =round(fit_inv_m.p00);
c10q =round(fit_inv_m.p10);
c20q =round(fit_inv_m.p20);
c30q =round(fit_inv_m.p30);
c01q =round(fit_inv_m.p01);
c11q =round(fit_inv_m.p11);
c21q =round(fit_inv_m.p21);
```

For the temperature calibration, the same methodology can be used. In such case the xt,yt matrices have to fitted in the following equation:

$$\mathrm{Tcal} = \mathrm{c0} * 0.5 + \mathrm{c1} * \mathrm{Traw}_{sc}$$

The matrices xt, yt are provided below:

4 Pressure Calibration flow

$$xt = \begin{bmatrix} T1(P1)/kT \\ T1(P2)/kT \\ T1(P3)/kT \\ T1(P4)/kT \\ T2(P1)/kT \\ T2(P2)/kT \\ T2(P3)/kT \end{bmatrix}, y = \begin{bmatrix} Tref1(Pref1) \\ Tref1(Pref2) \\ Tref1(Pref3) \\ Tref1(Pref4) \\ Tref2(Pref1) \\ Tref2(Pref2) \\ Tref2(Pref3) \end{bmatrix}$$

For the calculation of the c0,c1 coefficients, the test data have to be fitted to the formula. To obtain such coefficient estimates, the user can use the linear least squares method since the fitting polynomial is also linear. For simplicity we provide below an example of fitting code in Matlab where by default linear least squares method is used:

```
% Fitting
[fit_inv_tm,gof_tm] = fit(xt,yt,'poly1');

c1 = fit_inv_tm.p1;
c0 = fit_inv_tm.p2;

% Quantisation
c1q = round(fit_inv_tm.p1);
c0q = round(fit_inv_tm.p2*2^1);
```

5 ASIC Registers Map

Table 11 Registers Map

Description	Addr	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Comments
Pressure data	0x00	PRESSURE_DATA_OUT[23:16] (r)								
	0x01	PRESSURE_DATA_OUT[15:8] (r)								
	0x02	PRESSURE_DATA_OUT[7:9] (r)								
Temperature data	0x03	TEMPERATURE_DATA_OUT[23:16] (r)								
	0x04	TEMPERATURE_DATA_OUT[15:8] (r)								
	0x05	TEMPERATURE_DATA_OUT[7:0] (r)								
PRESSURE CONFIG	0x06	-	PRESSURE measurement rate, pm_rate [2:0] (rw)			Pressure measurement resolution pm_res[3:0] (rw)				
TEMP CONFIG	0x07	Temp Ext 0 - Internal 1 - External	Temperature measurement rate - tm_rate[2:0] (rw)			Temperature measurement resolution tm_res [3:0] (rw)				
Measurement Control Reg	0x08	Init_complete 1- Autosense has completed 0- Autosense in progress (r)	Trim_complete 1- Trimsense has completed 0 - Trimsense in progress (r)	Temp Data Ready 1- new data available (cleared when P result is read) (r)	Pressure Data Ready 1-new data available (cleared when P result is read) (r)	-	Measurement Control [2:0] 000- Idle/Stop Background 001 - Pressure Measurement 010-Temperature Measurement 011 - Not defined 100- Not defined 101- Continous Pressure Measurement 110 - Continous Temperature Measurement (rw)			

5 ASIC Registers Map

Table 11 Registers Map (continued)

Description	Addr	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Comments
Config Register	0x09	INT_H_L Interrupt Active Level: 0- Active low 1- Active High (Interrupt can be output on SDO pin when in I2C mode or 3-wire SPI) (rw)	INT_SEL[2:0] INT_SEL<0> 1..Pressure Interrupt enabled, 0..no P interrupt INT_SEL<1> 1..Temperature Interrupt enabled, 0.. no T interrupt INT_SEL<2> 1..FIFO full Interrupt enabled, 0.. no FIFO interrupt (rw)			T result shift enable 1..shift result right in 24bit reg 0..no shift (rw)	P result shift enable 1.. shift result right in 24bit reg 0..no shift (rw)	FIFO enable 1.. enable FIFO mode (read result by cycling through addr 0x00-0x03; 32x24bit values stored) 0..P/T result in separate registers (rw)	SPI Mode 0 - 4wire interface 1- 3-wire interface (rw)	
Interrupt Status/ Source Register (Bits clared when register is read)	0x0A	-	-	-	-	-	Int_FIFO_full 0 - No interrupt active 1- interrupt active (r)	Int-Temp 0 - No interrupt active 1 - interrupt active (r)	Int_Press 0 - No interrupt active 1 - interrupt active (r))	
FIFO status	0x0B	-	-	-	-	-	-	FIFO full 1...32 valid data values in FIFO (r)	FIFO empty 1.. No data in FIFO 0.. Valid data in FIFO (r)	

Table 11 Registers Map (continued)

Description	Addr	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Comments
RESET	0x0C	FIFO flush 1.. Empty FIFO (read pointer to write pointer). Do a flush after read out of FIFO in order to delete old data (rw)	-	-	-	Soft_reset[3:0] 1001 - Full soft reset - Will perform the same sequence as power on reset 0011 - SW reset of the Efuse latches - Used for Lab debugging - and overwrite of Efuse programming (this "reset" is triggered only, if a transition from 0000 -->0011 or 1001 is detected) (rw)				
Product ID - Revision # metal mask prorammable	0x0D	Rev_ID[7:4] (r)				Prod_ID[3:0] (r)				Reset value: 0x10hex
Signature	0x0E	"Hidden Register" Signature MSB <7:0> (w)								
	0x0F	"Hidden Register" Signature LSB <7:0> (w)								

5 ASIC Registers Map

Table 11 Registers Map (continued)

Description	Addr	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Comments
200 bit calibration data Every sensor module has individual coefficients. Before the first calculation of temperature and pressure, the master reads out the EFuse data. (only the right address is accesible in user space)	0x40 0x10	C0 <11:4> (rw)								
	0x41 0x11	C0 <3:0> (rw)				C1 <11:8> (rw)				
	0x42 0x12	C1<7:0> (rw)								
	0x43 0x13	C00 <19:12> (rw)								
	0x44 0x14	C00 <11:4> (rw)								
	0x45 0x15	C00<3:0> (rw)				C10<19:16> (rw)				
	0x46 0x16	C10 <15:8> (rw)								
	0x47 0x17	C10<7:0> (rw)								
	0x48 0x18	C01<15:8> (rw)								
	0x49 0x19	C01<7:0> (rw)								

5 ASIC Registers Map

Table 11 Registers Map (continued)

Description	Addr	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Comments
continue 200 bit calibration map	0x4A 0x1A	C11<15:8> (rw)								
	0x4B 0x1B	C11<7:0> (rw)								
	0x4C 0x1C	C20<15:8> (rw)								
	0x4D 0x1D	C20<7:0> (rw)								
	0x4E 0x1E	C21<15:8> (rw)								
	0x4F 0x1F	C21<7:0> (rw)								
	0x50 0x20	C30<15:8> (rw)								
	0x51 0x21	C30<7:0> (rw)								

Table 11 Registers Map (continued)

Description	Addr	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Comments
continue 200 bit calibration map	0x52 0x22	Calibration Register 18 <7:0> (rw)								
	0x53 0x23	Calibration Register 19 <7:0> (rw)								
	0x54 0x24	Calibration Register 20 <7:0> (rw)								
	0x55 0x25	Running Chip ID number <30:23> (rw)								
	0x56 0x26	Running Chip ID number <22:15> (rw)								
	0x57 0x27	Running Cchip ID number <14:7> (rw)								
	0x58 0x28	Temperature used in coeff set 0.. ASIC 1...MEMS	Running Chip ID number <6:0> (rw) (for identification of tester/lab samples only)							
1280KHz Osc Trim register	0x60 0x30									
LP (8KHz) Osc Trim register	0x61 0x31									

Table 11 Registers Map (continued)

Description	Addr	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Comments
Analog Config	0x62 0x32								Analog Conf <0> hi_gain for P measurement: 1 increases the ADC gain by a factor of 1.6 (=+4.08dB) (rw)	
LDO 1.5V Trim VCM08 Trim	0x63 0x33									
Amplifier adj	0x64 0x34	Amp_gain[3:0] Code Integration_Cap Gain 0 1525fF 0dB 1 1350fF 1.06dB 2 1200fF 2.08dB 3 1075fF 3.04dB (rw)								
MEMS Offset	0x65 0x35				MEMS Offset [4:0] Range: -100fF to +675fF, step 25fF , 0->100fF 8rw)					
Datapath Configuration	0x66 0x36									
I2C address	0x67 0x37									
I2C PHY Configuration	0x68 0x38									
Spare	0x69 0x39									

Table 11 Registers Map (continued)

Description	Addr	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Comments
Efuse_Fused	0x6F 0x3F								Efuse_Fused 1 - fused, signature enabled 0 - not finally fused, no signature needed to access hidden address space	
Efuse controls	0x70			2kIntClk 0 - Internal 8kHz Clock selected 1 - Internal 2kHz Clock selected (use the 2kHz clock, 8kHz is only fortest) (rw)	ExtClkSel 0 - Internal 8kHz Clock selected 1- External clock selected (CSB) (if 8kHz is not trimmed, the external clock should be used as accurate timebase) (rw)	-	TestMargin 0 - TestMargin off 1 - TestMargin on (rw)	Fuse Control <1:0> 00 -Idle 01 - Sense 10 - Blow 11 - Undefined (rw)		EFuse Control

6 Register Description

6.1 Pressure Data (PRS_Bn)

The Pressure Data registers contains the 24 bit (3 bytes) 2's complement pressure measurement value.
If the FIFO is enabled, the register will contain the FIFO pressure and/or temperature results. Otherwise, the register contains the pressure measurement results and will not be cleared after read.

6.1.1 PRS_B2

The highest byte of the three bytes measured pressure value.

PRS_B2 **Address:** 00_H
Pressure (MSB data) **Reset value:** 00_H

7	6	5	4	3	2	1	0
PRS23	**PRS22**	**PRS21**	**PRS20**	**PRS19**	**PRS18**	**PRS17**	**PRS16**

r

Field	Bits	Type	Description
PRS[23:16]	7:0	r	MSB of 24 bit 2´s complement pressure data.

6.1.2 PRS_B1

The middle byte of the three bytes measured pressure value.

PRS_B1 **Address:** 01_H
Pressure (LSB data) **Reset value:** 00_H

7	6	5	4	3	2	1	0
PRS15	**PRS14**	**PRS13**	**PRS12**	**PRS11**	**PRS10**	**PRS9**	**PRS8-**

r

Field	Bits	Type	Description
PRS[15:8]	7:0	r	LSB of 24 bit 2´s complement pressure data.

6.1.3 PRS_B0

The lowest byte of the three bytes measured pressure value.

PRS_B0	**Address:**	02_H
Pressure (XLSB data)	**Reset value:**	00_H

7	6	5	4	3	2	1	0
PRS7	**PRS6**	**PRS5**	**PRS4**	**PRS3**	**PRS2**	**PRS1**	**PRS0**

r

Field	Bits	Type	Description
PRS[7:0]	7:0	r	XLSB of 24 bit 2´s complement pressure data.

6.2 Temperature Data (TMP_Tn)

The Temperature Data registers contain the 24 bit (3 bytes) 2's complement temperature measurement value (unless the FIFO is enabled, please see FIFO operation) and will not be cleared after the read.

6.2.1 TMP_B2

The highest byte of the three bytes measured temperature value.

TMP_B2	**Address:**	03_H
Temperature (MSB data)	**Reset value:**	00_H

7	6	5	4	3	2	1	0
TMP23	**TMP22**	**TMP21**	**TMP20**	**TMP19**	**TMP18**	**TMP17**	**TMP16**

r

Field	Bits	Type	Description
TMP[23:16]	7:0	r	MSB of 24 bit 2´s complement temperature data.

6.2.2 TMP_B1

The middle byte of the three bytes measured temperature value.

TMP_B1	**Address:**	04_H
Temperature (LSB data)	**Reset value:**	00_H

7	6	5	4	3	2	1	0
TMP15	**TMP14**	**TMP13**	**TMP12**	**TMP11**	**TMP10**	**TMP9**	**TMP8**

r

Field	Bits	Type	Description
TMP[15:8]	7:0	r	LSB of 24 bit 2´s complement temperature data.

6.2.3 TMP_B0

The lowest part of the three bytes measured temperature value.

TMP_B0	**Address:**	05_H
Temperature (XLSB data)	**Reset value:**	00_H

7	6	5	4	3	2	1	0
TMP7	**TMP6**	**TMP5**	**TMP4**	**TMP3**	**TMP2**	**TMP1**	**TMP0**

r

Field	Bits	Type	Description
TMP[7:0]	7:0	r	XLSB of 24 bit 2´s complement temperature data.

6.3 Pressure Configuration (PRS_CFG)

Configuration of pressure measurement rate (PM_RATE) and resolution (PM_PRC).

PRS_CFG	**Address:**	06_H
Pressure measurement configuration	**Reset value:**	00_H

7	6	5	4	3	2	1	0
-	**PM_RATE[2:0]**			**PM_PRC[3:0]**			
-	rw			rw			

Field	Bits	Type	Description
-	7	-	Reserved.
PM_RATE[2:0]	6:4	rw	Pressure measurement rate: 000 - 1 measurements pr. sec. 001 - 2 measurements pr. sec. 010 - 4 measurements pr. sec. 011 - 8 measurements pr. sec. 100 - 16 measurements pr. sec. 101 - 32 measurements pr. sec. 110 - 64 measurements pr. sec. 111 - 128 measurements pr. sec. *Applicable for measurements in Background mode only*
PM_PRC[3:0]	3:0	rw	Pressure oversampling rate: 0000 - Single. (Low Precision) 0001 - 2 times (Low Power). 0010 - 4 times. 0011 - 8 times. 0100 *)- 16 times (Standard). 0101 *) - 32 times. 0110 *) - 64 times (High Precision). 0111 *) - 128 times. 1xxx - Reserved

*) Note: Use in combination with a bit shift. See ***Interrupt and FIFO configuration (CFG_REG)*** register

Table 12 Precision (Pa_{RMS}) and pressure measurement time (ms) versus oversampling rate

Oversampling (PRC[3:0])	**Single (0000)**	**2 times (0001)**	**4 times (0010)**	**8 times (0011)**	**16 times (0100)**	**32 times (0101)**	**64 times (0110)**	**128 times (0111)**
Measurement time (ms)	3.6	5.2	8.4	14.8	27.6	53.2	104.4	206.8
Precision (Pa_{RMS})	5		2.5		1.2	0.9	0.5	

Table 13 Estimated current consumption (uA)

Oversampling (PRC[3:0]) **Measurements pr sec. (PM_RATE([2:0])**	**Single (0000)**	**2 times (0001)**	**4 times (0010)**	**8 times (0011)**	**16 times (0100)**	**32 times (0101)**	**64 times (0110)**	**128 times (0111)**
1 (000)	2.1	2.7	3.8	6.1	11	20	38	75
2 (001)								
4 (010)								
8 (011)	*Note: The current consumption can be calculated as the Measurement Rate * Current Consumption of 1 measurement per. sec.*							n.a.
16 (100)							n.a.	n.a.
32 (101)						n.a.	n.a.	n.a.
64 (110)					n.a.	n.a.	n.a.	n.a.
128 (111)			n.a.	n.a.	n.a.	n.a.	n.a.	n.a.

Note: The table shows the possible combinations of Pressure Measurement Rate and oversampling when no temperature measurements are performed. When temperature measurements are performed the possible combinations are limited to $Rate_{temperature}$ x Measurement $Time_{temperature}$ + $Rate_{pressure}$ x Measurement $Time_{pressure}$ < 1 second.

The temperature measurement time versus temperature oversampling rate is similar with pressure measurement time versus pressure oversampling rate

6.4 Temperature Configuration(TMP_CFG)

Configuration of temperature measurement rate (TMP_RATE) and resolution (TMP_PRC).

TMP_CFG **Address:** 07_H

Temperature measurement configuration **Reset value:** 00_H

7	6	5	4	3	2	1	0
TMP_EXT	**TMP_RATE[6:4]**			**TMP_PRC[3:0]**			
rw	rw			-	rw		

Field	Bits	Type	Description
TMP_EXT	7	rw	Temperature measurement 0 - Internal sensor (in ASIC) 1 - External sensor (in pressure sensor MEMS element) *Note: It is highly recommended to use the same temperature sensor as the source of the calibration coefficients. Please see the* ***Coefficient Source*** *register*
TMP_RATE[2:0]	6:4	rw	Temperature measurement rate: 000 - 1 measurement pr. sec. 001 - 2 measurements pr. sec. 010 - 4 measurements pr. sec. 011 - 8 measurements pr. sec. 100 - 16 measurements pr. sec. 101 - 32 measurements pr. sec. 110 - 64 measurements pr. sec. 111 - 128 measurements pr. sec.. *Applicable for measurements in Background mode only*
TMP_PRC[2:0]	3:0	rw	Temperature oversampling (precision): 0000 - single. (Default) - Measurement time 3.6 ms. *Note: Following are optional, and may not be relevant:* 0001 - 2 times. 0010 - 4 times. 0011 - 8 times. 0100 - 16 times. 0101 - 32 times. 0110 - 64 times.. 0111 - 128 times. 1xxx - Reserved.

6.5 Sensor Operating Mode and Status (MEAS_CFG)

Setup measurement mode.

MEAS_CFG	**Address:**	08_H
Measurement configuration	**Reset value:**	$C0_H$

7	6	5	4	3	2	1	0
COEF_RDY	**SENSOR_RDY**	**TMP_RDY**	**PRS_RDY**	**-**	**MEAS_CTRL**		
r	r	r	r	-	rw		

Field	Bits	Type	Description
COEF_RDY	7	r	Coefficients will be read to the Coefficents Registers after start-up: 0 - Coefficients are not available yet. 1 - Coefficients are available.
SENSOR_RDY	6	r	The pressure sensor is running through self initialization after start-up. 0 - Sensor initialization not complete 1 - Sensor initialization complete It is recommend not to start measurements until the sensor has completed the self intialization.
TMP_RDY	5	r	Temperature measurement ready 1 - New temperature measurement is ready. Cleared when temperature measurement is read.
PRS_RDY	4	r	Pressure measurement ready 1 - New pressure measurement is ready. Cleared when pressurement measurement is read.
-	3	-	Reserved.
MEAS_CTRL	2:0	rw	Set measurement mode and type: *Standby Mode* 000 - Idle / Stop background measurement *Command Mode* 001 - Pressure measurement 010 - Temperature measurement 011 - na. 100 - na. *Background Mode* 101 - Continous pressure measurement 110 - Continous temperature measurement 111 - Continous pressure and temperature measurement

6.6 Interrupt and FIFO configuration (CFG_REG)

Configuration of interupts, measurement data shift, and FIFO enable.

CFG_REG	**Address:**	09_H
Configuration register	**Reset value:**	00_H

7	6	5	4	3	2	1	0
INT_HL	**INT_FIFO**	**INT_TMP**	**INT_PRS**	**T_SHIFT**	**P_SHIFT**	**FIFO_EN**	**SPI_MODE**
rw	rw	rw	rw	rw	rw	rw	rw

Field	Bits	Type	Description
INT_HL	7	rw	Interupt (on SDO pin) active level: 0 - Active low. 1 - Active high.
INT_FIFO	6	rw	Generate interupt when the FIFO is full: 0 - Disable. 1 - Enable.
INT_TMP	5	rw	Generate interupt when a temperature measurement is ready: 0 - Disable. 1 - Enable.
INT_PRS	4	rw	Generate interupt when a pressure measurement is ready: 0 - Disable. 1 - Enable.
T_SHIFT	3	rw	Temperature result bit-shift 0 - no shift. 1 - shift result right in data register. *Note: Must be set to '1' when the oversampling rate is >8 times.*
P_SHIFT	2	rw	Pressure result bit-shift 0 - no shift. 1 - shift result right in data register. *Note: Must be set to '1' when the oversampling rate is >8 times.*
FIFO_EN	1	rw	Enable the FIFO: 0 - Disable. 1 - Enable.
SPI_MODE	0	rw	Set SPI mode: 0 - 4-wire interface. 1 - 3-wire interface.

6.7 Interrupt Status (INT_STS)

Interrupt status register. The register is cleared on read.

INT_STS	**Address:**	$0A_H$
Interrupt status	**Reset value:**	00_H

7	6	5	4	3	2	1	0
-					**INT_FIFO_FULL**	**INT_TMP**	**INT_PRS**
-					r	r	r

Field	Bits	Type	Description
-	7:3	-	Reserved.
INT_FIFO_FULL	2	r	Status of FIFO interrupt 0 - Interrupt not active 1 - Interrupt active
INT_TMP	1	r	Status of temperature measurement interrupt 0 - Interrupt not active 1 - Interrupt active
INT_PRS	0	r	Status of pressure measurement interrupt 0 - Interrupt not active 1 - Interrupt active

6.8 FIFO Status (FIFO_STS)

FIFO status register

FIFO_STS **Address:** $0B_H$

FIFO status register **Reset value:** 00_H

7	6	5	4	3	2	1	0
-						**FIFO_FULL**	**FIFO_EMPTY**
-						r	r

Field	Bits	Type	Description
-	7:2	-	Reserved.
FIFO_FULL	1	r	0 - The FIFO is not full 1 - The FIFO is full
FIFO_EMPTY	0	r	0 - The FIFO is not empty 1 - The FIFO is empty

6.9 Soft Reset and FIFO flush (RESET)

Flush FIFO or generate soft reset.

RESET		**Address:**	$0C_H$
FIFO flush and soft reset		**Reset value:**	00_H

7	6	5	4	3	2	1	0
FIFO_FLUSH	**-**			**SOFT_RST**			
w	-			w			

Field	Bits	Type	Description
FIFO_FLUSH	7	w	FIFO flush 1 - Empty FIFO After reading out all data from the FIFO, write '1' to clear all old data.
-	6:4	-	Reserved.
SOFT_RST	3:0	w	Write '1001' to generate a soft reset. A soft reset will run though the same sequences as in power-on reset.

6.10 Product and Revision ID (ID)

Product and Revision ID.

ID		**Address:**	$0D_H$
Product and revision ID		**Reset value:**	$0x10_H$

7	6	5	4	3	2	1	0
REV_ID				**PROD_ID**			
r				r			

Field	Bits	Type	Description
REV_ID	7:4	r	Revision ID
PROD_ID	3:0	r	Product ID

6.11 Calibration Coefficients (COEF)

The Calibration Coefficients register contains the 2´s complement coefficients that are used to calculate the compensated pressure and temperature values.

Table 14 Calibration Coefficients

Coefficient	Addr.	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
c0	0x10	c0 [11:4]							
c0/c1	0x11	c0 [3:0]				c1 [11:8]			
c1	0x12	c1[7:0]							
c00	0x13	c00 [19:12]							
c00	0x14	c00 [11:4]							
c00/c10	0x15	c00 [3:0]				c10 [19:16]			
c10	0x16	c10 [15:8]							
c10	0x17	c10 [7:0]							
c01	0x18	c01 [15:8]							
c01	0x19	c01 [7:0]							
c11	0x1A	c11 [15:8]							
c11	0x1B	c11 [7:0]							
c20	0x1C	c20 [15:8]							
c20	0x1D	c20 [7:0]							
c21	0x1E	c21 [15:8]							
c21	0x1F	c21 [7:0]							
c30	0x20	c30 [15:8]							
c30	0x21	c30 [7:0]							

Note: Generate the decimal numbers out of the calibration coefficients registers data:

```
C20 := reg0x1D + reg0x1C * 2^ 8
if (C20 > (2^15 - 1))
  C20 := C20 - 2^16
end if

C0 := (reg0x10 * 2^ 4) + ((reg0x11 / 2^4) & 0x0F)
if (C0 > (2^11 - 1))
  C0 := C0 - 2^12
end if
```

6.12 Coefficient Source

States which internal temperature sensor the calibration coefficients are based on: the ASIC temperature sensor or the MEMS element temperature sensor. The coefficients are only valid for one sensor and it is highly recommended to use the same temperature sensor in the application. This is set-up in the Temperature Configuration register.

TMP_COEF_SRCE	**Address:**	28_H
Temperature Coefficients Source	**Reset value:**	XX_H

7	6	5	4	3	2	1	0
TMP_COEF_ SRCE	**-**						
r	-						

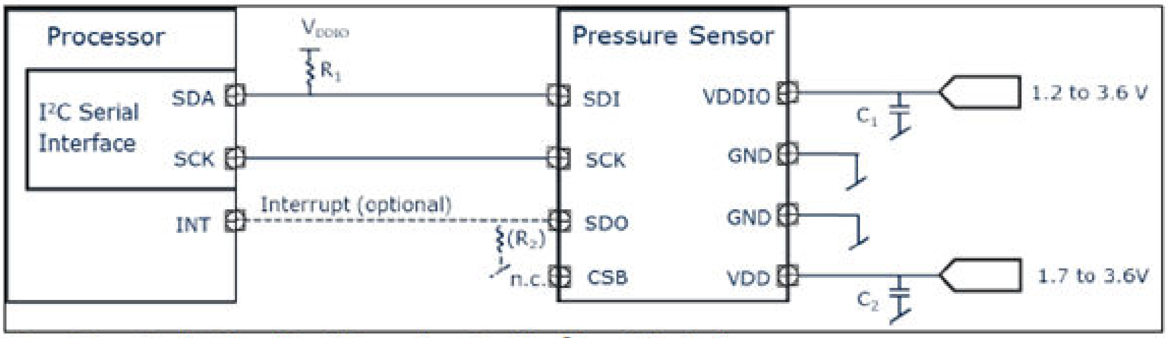
Field	Bits	Type	Description
TMP_COEF_SRCE	7	r	Temperature coefficients are based on: 0 - Internal temperature sensor (of ASIC) 1 - External temperature sensor (of pressure sensor MEMS element)
-	6:0	-	Reserved

7 Appendix A. Digital Pressure Sensor (DPS) quick start

I. Power up

For the application example circuitry see the datasheet or below figures:

The example application circuit example uses the I²C serial interface. The *SDO* pin can be used for interrupt or to set least significant bit of the device address.



Component	Symbol	Values			Unit	Note / Test Condition
		Min.	Typ.	Max.		
Pull-up/down Resistor	R_1, R_2	5		100	KΩ	R_2 is optional and will set the address to 0x76 instead of 0x77.
Supply Blocking Capacitor	C_1, C_2	100	100		nF	The blocking capacitors should be placed as close to the package pins as possible.

The DPS starts up when VDD>=1,7V and VDDIO >= 1,2V is connected. Both supplies may also be connected together, eg. to a common 1,8V. Power up is finished when COEF_RDY=='1' and SENSOR_RDY=='1':

MEAS_CFG — Address: 08_H

Measurement configuration — Reset value: 00_H

7	6	5	4	3	2	1	0
COEF_RDY	SENSOR_RDY	TMP_RDY	PRS_RDY	-	MEAS_CTRL		
r	r	r	r	-	rw		

7 Appendix A. Digital Pressure Sensor (DPS) quick start

These bits need not to be read, if it can be ensured, that after connecting the supplies the chip is not accessed before a wait time of 40ms.

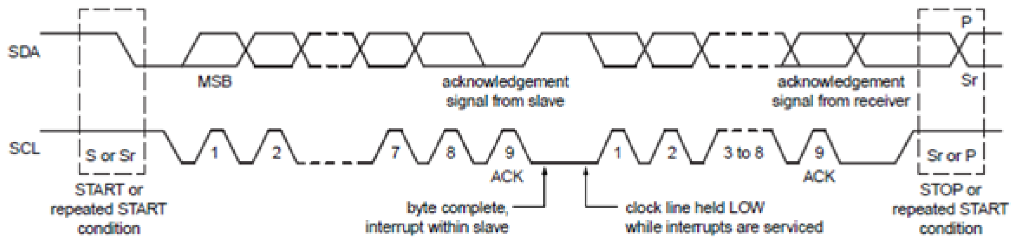
II. Interfacing (I2C)

The I2C address is 0x77 (SDO unconnected or connected to VDDIO), or 0x76 (SDO connected to GND).

The I2C protocol follows the standard, and supports the normal mode, FS, FS+ and HS mode:

http://www.nxp.com/documents/user_manual/UM10204.pdf

The basic timing is shown in the figure below (source UM10204.pdf) :



In one access (without stop) incremental read (address is auto increment) and incremental write (autoincrement) is supported. The read and write access is described below:

write command to DPS310

S	Module Address	W	A	Register Address	A	Register Data <7:0>	A	P	

write DPS310 register, autoincrement

S	Module Address	R	A	Register Address	A	Register Data 1 <7:0>	A	......	Register Data N <7:0>	A	P	

read DPS310 register

S	Module Address	R	A	Register Address	A	Register Data	A	N	P

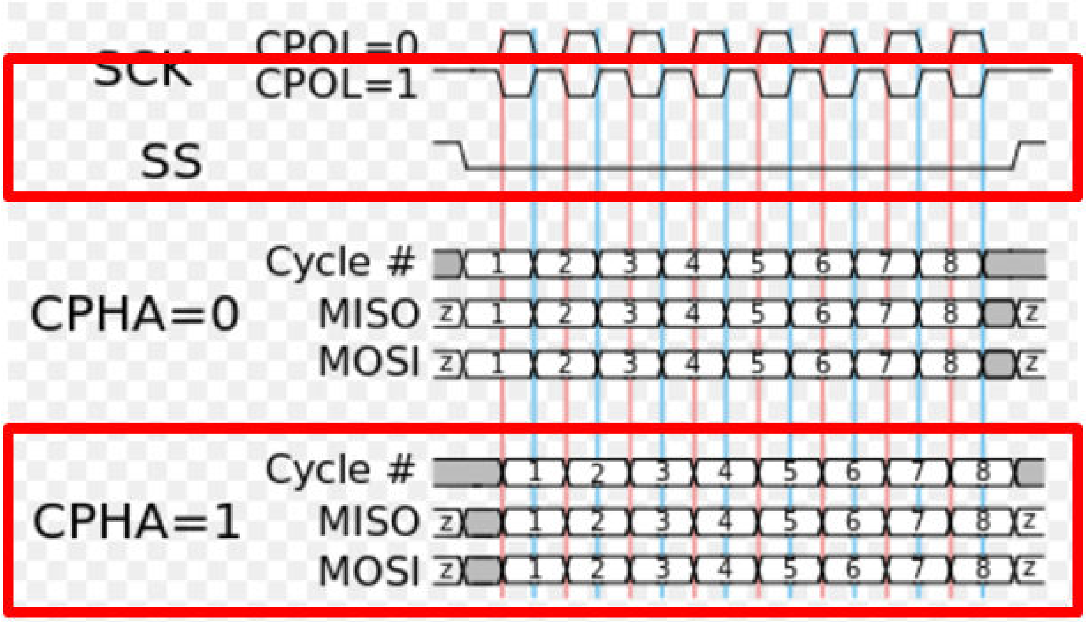
read DPS310 register, autoincrement

S	Module Address	R	A	Register Address	A	Register1 <7:0>	A	......	RegisterN <7:0>	N	P

S	start
P	stop
W	write
R	read
A	Acknowledge
N	Not Acknowledge
	Master
	Slave

III. Interfacing (SPI)

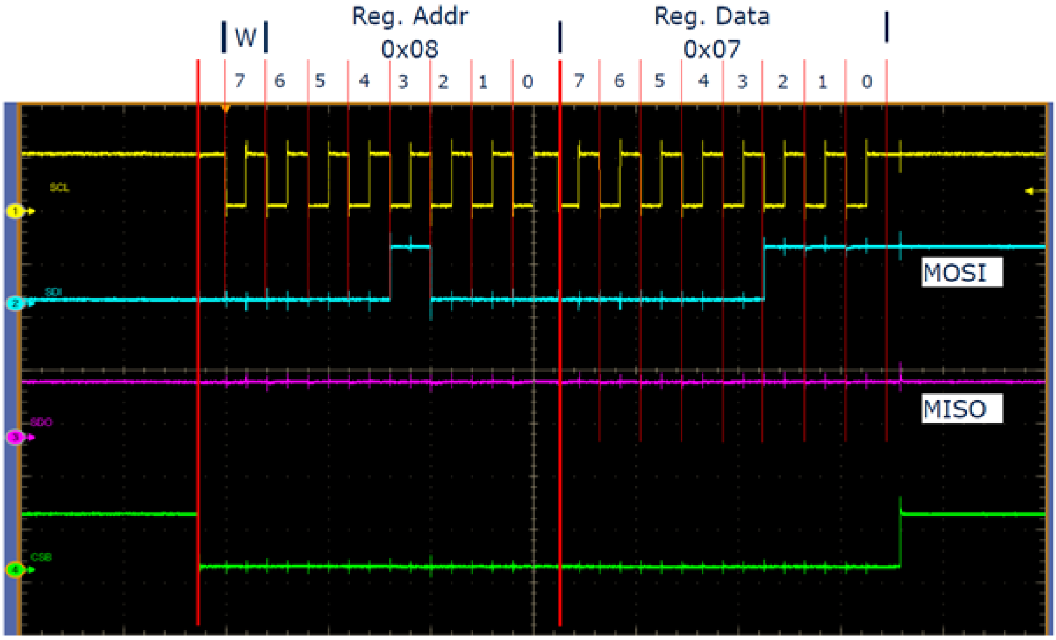
SPI Mode 3 (CPOL=1, CPHA=1) is supported. Clock speed up to 10MHz. When reading from the device, the returned data address is automatically incremented as long as the Chip Select (SS) is low.



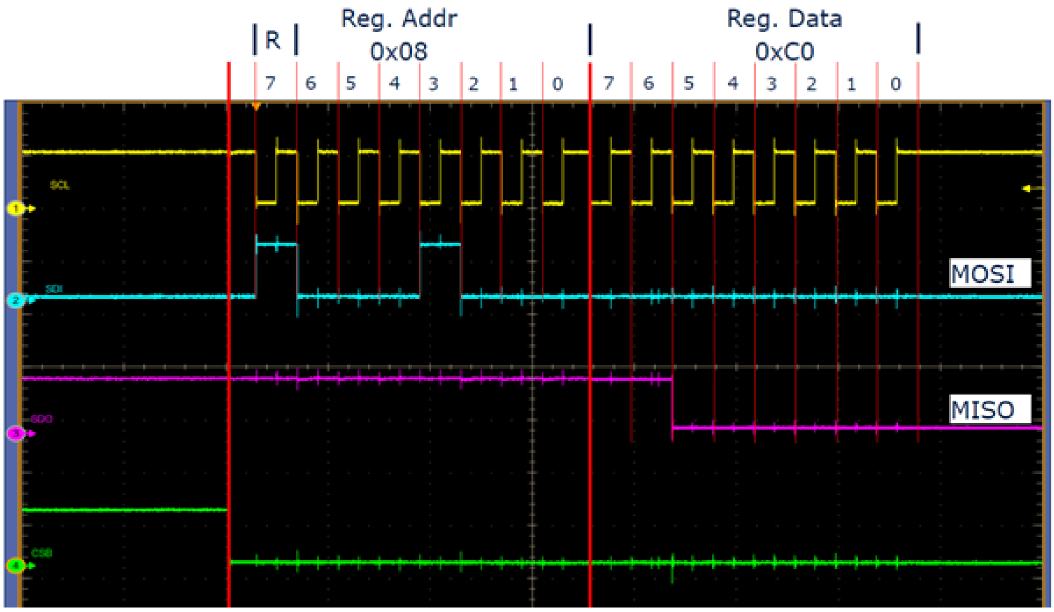
(source https://de.wikipedia.org/wiki/Serial_Peripheral_Interface#/media/File:SPI_timing_diagram2.svg)

7 Appendix A. Digital Pressure Sensor (DPS) quick start

Example SPI Write (first bit is W=0/R=1):



Example SPI Read (first bit is W=0/R=1):



IV. Read the DPS coefficients

Read register 0x10 – 0x20 to get the coefficients. The coefficients are stored as two's complement numbers.

7 Appendix A. Digital Pressure Sensor (DPS) quick start

C0 and C1 are 12 bit two's complement numbers.
C01, C11, C20, C21,C30 are 16 bit two's complement numbers.
C00, C10 are 20 bit two's complement numbers.

The coefficients are stored like described below:

Coefficient	Addr.	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
c0	0x10	c0 [11:4]							
c0/c1	0x11	c0 [3:0]				c1 [11:8]			
c1	0x12	c1[7:0]							
c00	0x13	c00 [19:12]							
c00	0x14	c00 [11:4]							
c00/c10	0x15	c00 [3:0]				c10 [19:16]			
c10	0x16	c10 [15:8]							
c10	0x17	c10 [7:0]							
c01	0x18	c01 [15:8]							
c01	0x19	c01 [7:0]							
c11	0x1A	c11 [15:8]							
c11	0x1B	c11 [7:0]							
c20	0x1C	c20 [15:8]							
c20	0x1D	c20 [7:0]							
c21	0x1E	c21 [15:8]							
c21	0x1F	c21 [7:0]							
c30	0x20	c30 [15:8]							
c30	0x21	c30 [7:0]							

Generate the decimal numbers out of the "raw coefficients".

```
e.g.:
C20= reg0x1D+ reg0x1C<8;
If C20 > (2^15-1)
        C20=C20-2^16;

C0= reg0x10<<4+(reg0x11>>4)&0x0F;
If C0 > (2^11-1)
        C0=C0-2^12;
```

V. Read from where the temperature is read

Read the Coefficient source register. States which internal temperature sensor the calibration coefficients are based on: the ASIC temperature sensor or the MEMS element temperature sensor. The coefficients are only valid for one sensor and it is highly recommended to use the same temperature sensor in the application.

TMP_COEF_SRCE — Address: 28_H

Temperature Coefficients Source — Reset value: XX_H

7	6	5	4	3	2	1	0
TMP_COEF_SRCE	-						
r	-						

Field	Bits	Type	Description
TMP_COEF_SRCE	7	r	Temperature coefficients are based on: 0 - Internal temperature sensor (of ASIC) 1 - External temperature sensor (of pressure sensor MEMS element)
-	6:0	-	Reserved

If TMP_COEF_SRCE == ‘1’ then

SELTMP=0x80

Else

SELTMP=0x00

See in next chapter how this info is used when the register 0x07 (TMP_CFG) is configured

VI. Configure the measurement accuracy of DPS

Some register settings must be done – depending on the expected operation in standard, high precision or low power mode. Note, that for higher precision a shift has to be programmed into register 0x09.

For three different configurations examples are given:

low power	average 512 (precision: two times)	
ADR	Val	
0x06	0x01	P config
0x07	0x00 | SELTMP	T config

standard	average 4096 (precision: 16 times)	
ADR	Val	
0x06	0x04	P config
0x07	0x00 | SELTMP	T config
0x09	0x04	P shift config

high precision	average 16384 (precision: 64 times)	
ADR	Val	
0x06	0x06	P config
0x07	0x00 | SELTMP	T config
0x09	0x04	P shift config

7 Appendix A. Digital Pressure Sensor (DPS) quick start

VII. Start the Pressure Measurement

Write to register 0x08 to start a measurement:

ADR	Val	
0x08	0x01	P start

VIII. Read the Pressure Result

1.) Wait for some time (depending on accuracy, 100ms is always safe)

OR

2.) Check if result is available: read register 0x08, check if PRS_RDY (bit 4) == 1

MEAS_CFG — Address: 08_H

Measurement configuration — Reset value: 00_H

7	6	5	4	3	2	1	0
COEF_RDY	SENSOR_RDY	TMP_RDY	PRS_RDY	-	MEAS_CTRL		
r	r	r	r	-	rw		

3.) Read register 0x00 -0x02 for the result

Register Name	Addr.	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PSR_B2	0x00	PSR[23:16] (r)							
PSR_B1	0x01	PSR[15:8](r)							
PSR_B0	0x02	PSR[7:0](r)							
TMP_B2	0x03	TMP[23:16] (r)							
TMP_B1	0x04	TMP[15:8] (r)							
TMP_B0	0x05	TMP[7:0] (r)							

Generate a Praw out of the result

(result is stored in registers as 24 bit 2s complement number).

e.g.:

```
Praw= reg0x02+ reg0x01<<8+reg0x00<<16;
If Praw > (2^23-1)
        Praw=Praw-2^24;
```

7 Appendix A. Digital Pressure Sensor (DPS) quick start

IX. Start the Temperature Measurement

Write to register 0x08 to start a measurement:

ADR	Val	
0x08	0x02	T start

X. Read the Temperature Result

1.) Wait for some time (depending on accuracy, 100ms is always safe)

OR

2.) Check if result is available: read register 0x08, check if Temp Data Ready (bit 5) == 1

MEAS_CFG **Address:** 08_H

Measurement configuration **Reset value:** 00_H

7	6	5	4	3	2	1	0
COEF_RDY	SENSOR_RDY	TMP_RDY	PRS_RDY	-	MEAS_CTRL		
r	r	r	r	-	rw		

3.) Read register 0x03 -0x05 for the result

Register Name	Addr.	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PSR_B2	0x00	PSR[23:16] (r)							
PSR_B1	0x01	PSR[15:8](r)							
PSR_B0	0x02	PSR[7:0](r)							
TMP_B2	0x03	TMP[23:16] (r)							
TMP_B1	0x04	TMP[15:8] (r)							
TMP_B0	0x05	TMP[7:0] (r)							

Generate a Traw out of the result

(result is stored in registers as 24 bit 2s complement number).

e.g.:

```
Traw= reg0x05+ reg0x04<<8+reg0x03<<16;
If Traw > (2^23-1)
        Traw=Traw-2^24;
```

7 Appendix A. Digital Pressure Sensor (DPS) quick start

XI. Calculate the physical Pressure and Temperature

Praw_sc= Praw/kP; *(Praw… 24bit two's complement value out of reg 0-2)*

Traw_sc= Traw/kT; *(Traw… 24bit two's complement value out of reg 3-5)*

kP and kT can be looked up in a reference table.
In the above configuration example the kP or kT are:

average	kP or kT	
16384	1040384	kP (high Precision)
12288	778240	
8192	516096	
6144	385024	
4096	253952	kP (Standard)
3072	188416	
2048	7864320	
1536	5767168	
1024	3670016	
768	2621440	
512	1572864	kP (Low Power)
384	1048576	
256	524288	kT (Low Power/Standard/high Precision)

Pcomp(Pa) = c00 + Praw_sc*(c10 + Praw_sc *(c20+ Praw_sc *c30)) + Traw_sc*c01 + Traw_sc *Praw_sc *(c11+Praw_sc*c21)

Tcomp (°C) = c0*0.5 + c1*Traw_sc

XII. Register reference

Register Name	Addr.	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Reset State
PSR_B2	0x00	PSR[23:16] (r)								00_h
PSR_B1	0x01	PSR[15:8](r)								00_h
PSR_B0	0x02	PSR[7:0](r)								00_h
TMP_B2	0x03	TMP[23:16] (r)								00_h
TMP_B1	0x04	TMP[15:8] (r)								00_h
TMP_B0	0x05	TMP[7:0] (r)								00_h
PRS_CFG	0x06	-	PM_RATE [2:0] (rw)			PM_PRC [3:0] (rw)				00_h
TMP_CFG	0x07	TMP_ EXT (rw)	TMP_RATE [2:0] (rw)			TM_PRC [3:0] (rw)				00_h
MEAS_CFG	0x08	COEF_ RDY (r)	SENSOR _RDY (r)	TMP_ RDY (r)	PRS_ RDY (r)	-	MEAS_CRTL [2:0] (rw)			00_h
CFG_REG	0x09	INT_ HL (rw)	INT_ SEL [2:0] (rw)			TMP_ SHIFT_ EN (rw)	PRS_ SHIFT_ EN (rw)	FIFO_ EN (rw)	SPI_ MODE (rw)	00_h
INT_STS	0x0A	-	-	-	-	-	INT_ FIFO_ FULL (r)	INT_ TMP(r)	INT_ PRS(r)	00_h
FIFO_STS	0x0B	-	-	-	-	-	-	FIFO_ FULL(r)	FIFO_ EMPTY(r)	00_h
RESET	0x0C	FIFO_ FLUSH (w)	-	-	-	SOFT_RST [3:0] (w)				00_h
ID	0x0D	PROD_ID [3:0] (r)				REV_ID [3:0] (r)				00_h
COEF	0x10-0x21	< see register description >								XX_h
Reserved	0x22-0x27	Reserved								XX_h
COEF_SRCE	0x28	TMP_COEF_SRCE (r)	Reserved							XX_h

8 Revision History

Major changes since previous revision

Table 15 Revision History

Reference	
v0.1	first released
v0.2	BE_fuse update
v0.3	BE_fuse updated VDD=4.1V when fusing; eFuse map bit 1 = 0 on the address 0x62 when fusing e-trimmed ASIC ; Register map, CFG_REG
v0.4	BE_fuse update
v0.5	BE_fuse update
v0.6	BE_fusechip_default and BE_gain update

Edition 2016-8-11

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about any aspect of this document?

Email: erratum@infineon.com

Document reference
IFX-poe1436167344349